



AF/2700

PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of

Docket No: ST9-99-179/A8496

MANDEL III, Richard H.

Appln. No.: 09/545,592

Group Art Unit: 2172

Confirmation No.: 9939

Examiner: Tam V. Nguyen

Filed: April 7, 2000

RECEIVED

For: . CROSS-PLATFORM SUBSELECT METADATA EXTRACTION

JUL 31 2003

SUBMISSION OF APPELLANT'S BRIEF ON APPEAL Technology Center 2100

MAIL STOP APPEAL BRIEF - PATENTS

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Submitted herewith please find an original and two copies of Appellant's Brief on Appeal. A check for the statutory fee of \$320.00 is attached. The USPTO is directed and authorized to charge all required fees, except for the Issue Fee and the Publication Fee, to Deposit Account No. 19-4880. Please also credit any overpayments to said Deposit Account. A duplicate copy of this paper is attached.

Respectfully submitted,

Billy Catm Raul
for J. Warren Lytle, Jr. REG. NO. 52,156
Registration No. 39,283

SUGHRUE MION, PLLC
Telephone: (202) 293-7060
Facsimile: (202) 293-7860

WASHINGTON OFFICE



23373

PATENT TRADEMARK OFFICE

Date: July 28, 2003

#14
Wm
8/8/03



PATENT APPLICATION

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

In re application of

Docket No: ST9-99-179/A8496

MANDEL III, Richard H.

Appln. No.: 09/545,592

Group Art Unit: 2172

Confirmation No.: 9939

Examiner: Tam V. Nguyen

Filed: April 7, 2000

RECEIVED

For: CROSS-PLATFORM SUBSELECT METADATA EXTRACTION

JUL 31 2003

APPELLANTS' BRIEF ON APPEAL UNDER 37 C.F.R. § 1.192

Technology Center 2100

MAIL STOP APPEAL BRIEF - PATENTS

Commissioner for Patents

P.O. Box 1450

Alexandria, VA 22313-1450

Sir:

In accordance with the provisions of 37 C.F.R. § 1.192, Appellant submits the following:

I. REAL PARTY IN INTEREST

The real party in interest in this appeal is International Business Machines Corporation ("IBM") of Armonk, New York, the assignee.

II. RELATED APPEALS AND INTERFERENCES

There are no other appeals or interferences known to Appellant, Appellant's legal representative, or the assignee that will directly affect or be directly affected by, or have a bearing on, the Board's decision in this appeal.

III. STATUS OF CLAIMS

Claims 1 - 33 are the claims pending in the application and are the subject of this appeal. None of the claims have been amended. A copy of the claims on appeal is set forth in an attached Appendix.

IV. STATUS OF AMENDMENTS

All responses to Office Actions submitted to the USPTO have been entered. A Response Under 37 C.F.R. § 1.116 was filed on March 27, 2003, in response to a final Office Action (Paper No. 6), and the Examiner indicates in the Advisory Action (Paper No. 11) that the Response was considered and entered.

V. SUMMARY OF THE INVENTION

The application on appeal describes techniques for obtaining metadata from databases that operate on different computing platforms. See pg. 8, line 19. In a relational database, for example, data records are stored in tables that consist of multiple rows and columns of data. Metadata information refers to information about other information. See page 2, line 1. Examples of metadata include the column names for the columns in a database table, or the type of data stored in a column. A result set is obtained when executing a query against a database and can contain various types of data. It is desirable and useful to know what types of data are contained in the result set. See pg. 2, line 14. Some computing platforms provide a predefined command designed to return metadata. For example, in the IBM DB2 Universal Database system, a DESCRIBE command will list column names and data types for a result set. See pg. 1,

lines 20-24. However, not all platforms provide such a command to return metadata.

Accordingly, there is a need to provide a technique for obtaining metadata information across various types of computing platforms. See pg. 2, lines 14-15.

The inventive techniques described in the application allow metadata to be retrieved from a result set even when the query producing the result set operates on a computing platform that does not have a command for returning such metadata. A query language, such as the Structured Query Language (SQL), is used to create queries to search a database. The queries, when executed, produce a result set from the database.

Fig. 3 of the application illustrates an inventive technique for obtaining metadata from a result set generated by a platform that does not have a database command for returning metadata. Here, a query statement, such as an SQL SELECT statement contains certain clauses (i.e., WHERE and GROUPBY clauses). The statement is modified to replace those WHERE and GROUPBY clauses with a logically false statement. For example the SELECT statement is modified to include a logically false statement such as WHERE 1 = -1. In this example 1 can never equal -1 and therefore the WHERE clause is evaluated to be a false condition. See pg. 10, lines 26-27. By modifying the SQL clause to include a false condition no data is returned when the clause is executed. This is because no row in the result set can satisfy the logically false condition. However, metadata for any data that would be returned if the false condition were not present, is returned. See pg. 11, lines 1-4.

Accordingly, by modifying a query statement, such as an SQL statement, to include a false clause and executing the modified query, metadata can be returned even in computing platforms that do not have a metadata extraction command.

VI. ISSUES

The issues on appeal are set forth as follows.

1) Whether claims 1-3, 6-14, 17-25 and 28-33 are unpatentable under 35 U.S.C. § 103(a) as being obvious over U.S. Patent No. 6,430,556 to Goldberg et al. ("Goldberg") in view of U.S. Patent No. 5,339,419 to Chan et al. ("Chan").

2) Whether claims 4-5, 15-16 and 26-27 are unpatentable under 35 U.S.C. § 103(a) as being obvious over Goldberg in view of U.S. Patent No. 6,199,063 to Colby et al. ("Colby").

VII. GROUPING OF CLAIMS

The claims do not stand or fall together and arguments for patentability of each group of claims, identified below, are set forth in this brief.

Group I: 1-3, 8-14, 19-25, and 30-33, each of which stand or fall together.

Group II: 6-7, 17-18, and 28-29, each of which stand or fall together.

Group III: 4-5, 5-16, and 26-27, each of which stand or fall together.

VIII. ARGUMENTS

Appellant respectfully requests the Board to reverse the Examiner's final rejections of the claims pending in the application for the following reasons.

APPELLANTS' BRIEF ON APPEAL
UNDER 37 C.F.R. § 1.192
U.S. Appln. No.: 09/545,592

A. The Prior Art

Each of the prior art references upon which the Examiner relies to reject the claims is discussed below.

Goldberg relates to a "visual integrated development environment for generating, installing, initializing and testing query objects which can operate with different databases while presenting a consistent interface to the user," (col. 3, lines 31-35). Goldberg's development environment includes a query object generator tool that generates query objects. Those query objects contain database queries and a graphical user interface (GUI) that a software developer uses to control the generator tool and build and test queries. See Abstract. A feature supported by the integrated development environment is a "data schema access query object" that retrieves metadata for display and to assist the developer in constructing a query object. See col. 3, lines 47-53. Goldberg does not disclose how the data schema access query object obtains the metadata. The data schema access query object also converts a query to a vendor-specific language used by a particular database and submits the query to the database engine. See col. 3, lines 56-60. Goldberg also discloses displaying the query objects on a graphic display so the developer can use various menu options to manipulate the queries such as by adding, deleting or modifying queries contained within the query object. See col. 10, lines 31-38.

Chan is directed to a computer software compiler system that distributes a machine-independent computer program. Chan discloses generating and distributing an application program in compiler-intermediate code, preferably HPcode-Plus, which is an architecture independent intermediate code. See col. 12, line 45-57. The portion of Chan the Examiner relies

APPELLANTS' BRIEF ON APPEAL
UNDER 37 C.F.R. § 1.192
U.S. Appln. No.: 09/545,592

upon in the final Office Action describes a conditional evaluation expression of the HPcode-Plus compiler intermediate language. See col. 19, lines 11-34. Here, Chan discloses the HPcode-Plus instruction referred to as a "Conditional Evaluate Expression" (CEXP) instruction that evaluates a boolean expression. If the result is true, then a true clause consisting of a first set of instructions is executed (the instructions up to the CSEP instruction associated with the CEXP instruction). If the result is false, then a false clause consisting of a second set of instructions is executed (the instructions from the CSEP instruction to the CEND instruction associated with the CEXP instruction). Accordingly, the conditional evaluation instruction CEXP has associated with it a true clause that is executed if the CEXP instruction evaluates true, and a false clause that is executed if the CEXP instruction evaluates false. Chan discloses that executing either the true or false clause results in zero or one item being pushed onto an expression stack. Col. 19, lines 17-19.

Colby relates to a system for rewriting a query in such a manner that a precomputed database view can be used to satisfy the original query in order to derive the answer to the query in the most efficient manner. See Abstract. Colby discloses several types of query clauses that are used in the rewritten queries for the precomputed tables, including: "select," "from," "where," and "groupby." Col. 8, lines 8-27.

APPELLANTS' BRIEF ON APPEAL
UNDER 37 C.F.R. § 1.192
U.S. Appln. No.: 09/545,592

B. Group I: Even If Goldberg Was Modified Based On Chan, The Combination Would Not Meet All The Limitations Of The Claims In Group I.

Claims 1-3, 8-14, 19-25, and 30-33 (Group I) are rejected under 35 U.S.C. § 103(a) as being unpatentable over Goldberg in view of Chan. Appellant respectfully submits that it would not have been obvious to modify the teachings of Goldberg with the teachings of Chan, but even if they were modified as the Examiner asserts, all the limitations of the claims would not be satisfied.

Claim 1, for example, is directed to a method for executing a query against a database on a data storage device connected to a computer. The claim recites modifying the query to replace one or more selected clauses with a false clause, executing the modified query with the false clause, and retrieving metadata from the result set obtained by executing the modified query.

In the final Office Action the Examiner relies on Goldberg for disclosing all the limitations recited in the independent claims except for teaching the use of a false clause. The Examiner admits that Goldberg does not disclose modifying a query to replace one or more selected clauses with a false clause and relies on Chan for disclosing a false clause. The Examiner asserts that it would have been obvious to combine the teachings of Goldman relating to modifying a query, with the Chan's use of a false clause in a conditional expression.

In the final Office Action, at page 3, the Examiner appears to take the position that because the true and false clauses of Chan's conditional evaluation instruction result in "zero or one item being pushed onto the expression stack," that this would somehow motivate a person of ordinary skill in the art modify Goldberg to use a false clause as in Chan's conditional evaluation instruction. In the Advisory Action (Paper No. 11) the Examiner asserts for the first

APPELLANTS' BRIEF ON APPEAL
UNDER 37 C.F.R. § 1.192
U.S. Appln. No.: 09/545,592

time that a person of ordinary skill in the art would be motivated to combine the teachings of the references because the combination allegedly would provide an architecture neutral distribution format (ANDF) that can operate on any computer platform. However, even if Goldman were modified to use the HPcode-Plus compiler intermediate language disclosed in Chan, there is no teaching or suggestion that using Chan's compiler-intermediate language would result in modifying a query to replace one or more selected clauses with the Chan's CEXP instruction that includes a false clause, as required by claim 1. The fact that Chan discloses a conditional evaluation instruction CEXP that includes a false clause does not mean that the CEXP instruction would be used to modify a query. Neither Goldman nor Chan teaches or suggests that such a conditional expression CEXP would be used in any modified query, much less in modifying a query to replace one or more selected clauses with the CEXP instruction.

Accordingly, even if Goldberg were modified to use the HPcode-Plus intermediate code instructions of Chan, there is no teaching or suggestion that such a modification would use the false clause in the conditional evaluation expression CEXP to replace selected clauses in the queries of Goldberg. Hence, the asserted combination does not meet all the limitations of the claim 1.

The remaining claims in Group I contain similar limitations, and hence, are patentable for at least the same reasons.

C. Group II: Goldberg and Chan, either alone or in combination, do not teach or suggest generating a list of modified queries, wherein each modified query has one or more selected clauses replaced with a false clause.

The Examiner, rejects claims 6-7, 17-18, and 28-29 (Group II) under 35 U.S.C. § 103(a) as being unpatenable over Goldberg in view of Chan. Applicant respectfully traverses the rejection and requests reconsideration for at least the following reasons.

Claim 6, for example, recites “generating a list of modified queries, wherein each modified query has one or more selected clauses replaced with a false clause; and executing each modified query until one executes successfully.”

The Examiner, in the final Office Action, refers to col. 2, line 36 through col. 3, line 55, and col. 10, lines 3-38 of Goldberg in rejecting the claims in Group II. However, Goldman neither teaches nor suggests generating a list of modified queries, much less “executing each until one executes successfully,” as required by the claims of Group II. Although Goldberg, at col. 10, lines 31-38, indicates that each query object contains one or more queries, and that the “Modify Query” function can operate on those queries in the query object, there is no teaching or suggestion that queries in a list are executed until one executes successfully, as required by the claims in Group II.

Chan, considered alone or in combination with Goldberg, also does not teach or suggest that limitation of claim 6.

Accordingly, it is respectfully submitted that even if the teachings of Goldberg were combined with Chan, as the Examiner asserts, all the limitations of claim 6 and the other claims in Group II would not be satisfied.

D. Group III: Even If Goldberg Was Modified Based On Colby, The Combination Would Not Meet All The Limitations Of The Claims In Group III

Claims 4-5, 15-16 and 26-27 (Group III) are rejected under 35 U.S.C. §103(a) as being unpatentable over Goldberg in view of Colby et al. ("Colby").

Claims 4, 15 and 26 specify the selected clauses recited in the independent claims as WHERE clauses. Claims 5, 16 and 27 specify the selected clauses recited in the independent claims as GROUPBY clauses. The Examiner relies on Colby for disclosing various query clauses such as "select", "from", "where" and "group by." At page 5 of the final Office Action the Examiner asserts that it would have been obvious to modify Goldberg to replace "where" and "groupby" clauses based on Colby's disclosure of those clauses. The motivation identified by the Examiner for modifying Goldberg is to use the materialized views disclosed in Colby for increased speed and efficiency in deriving an answer to a query.

It is respectfully submitted that Colby does not satisfy the deficiencies of the Goldberg/Chan combination discussed above. Further, although Colby discloses select and groupby query clauses, neither Goldman nor Colby, either alone or in combination teaches or suggests replacing a WHERE clause or a GROUPBY clause with a false clause. Even if Goldman were modified to rewrite queries in order to use the precomputed views of Colby, there is no teaching or suggestion that a WHERE clause or a GROUPBY clause would be replaced with a false clause to use the precompute views. Accordingly, it is respectfully submitted that a Goldman/Colby combination does not render claims 4-5, 15-16 and 26-27 unpatentable.

APPELLANTS' BRIEF ON APPEAL
UNDER 37 C.F.R. § 1.192
U.S. Appln. No.: 09/545,592

The present Brief on Appeal is being filed in triplicate. Unless a check is submitted herewith for the fee required under 37 C.F.R. §1.192(a) and 1.17(c), please charge said fee to Deposit Account No. 19-4880.

The USPTO is directed and authorized to charge all required fees, except for the Issue Fee and the Publication Fee, to Deposit Account No. 19-4880. Please also credit any overpayments to said Deposit Account.

Respectfully submitted,

Billy Cath. Rauh
for J. Warren Lytle, Jr. *Reg. No. 52,156*
Registration No. 39,283

SUGHRUE MION, PLLC
Telephone: (202) 293-7060
Facsimile: (202) 293-7860

WASHINGTON OFFICE



23373

PATENT TRADEMARK OFFICE

Date: July 28, 2003

APPENDIX

CLAIMS 1-33 ON APPEAL:

1. A method for executing a query against a database on a data storage device connected to a computer, the method comprising:
modifying the query to replace one or more selected clauses with a false clause;
executing the modified query with the false clause; and
retrieving metadata from the result set obtained by executing the modified query.
2. The method of claim 1, wherein the query comprises a SELECT statement.
3. The method of claim 2, wherein the SELECT statement is not a SELECT INTO statement.
4. The method of claim 1, wherein the selected clauses comprise WHERE clauses.
5. The method of claim 1, wherein the selected clauses comprise GROUP BY clauses.
6. The method of claim 1, wherein modifying the query comprises:
generating a list of modified queries, wherein each modified query has one or more selected clauses replaced with a false clause; and
executing each modified query until one executes successfully.
7. The method of claim 6, wherein a query executes successfully if it executes without an exception.

APPELLANTS' BRIEF ON APPEAL
UNDER 37 C.F.R. § 1.192
U.S. Appln. No.: 09/545,592

8. The method of claim 1, wherein the metadata comprises column type data for the result set.
9. The method of claim 8, further comprising converting the column type data to JAVA types.
10. The method of claim 9, further comprising generating a SQLJ iterator with parameters having the JAVA types.
11. The method of claim 1, further comprising determining whether the query requires a SQLJ iterator.
12. An apparatus for executing a query, comprising:
a computer connected a data storage device that stores a database containing data;
one or more computer programs, performed by the computer, for modifying the query to replace one or more selected clauses with a false clause, executing the modified query with the false clause, and retrieving metadata from the result set obtained by executing the modified query.
13. The apparatus of claim 12, wherein the query comprises a SELECT statement.
14. The apparatus of claim 13, wherein the SELECT statement is not a SELECT INTO statement.
15. The apparatus of claim 12, wherein the selected clauses comprise WHERE clauses.

APPELLANTS' BRIEF ON APPEAL
UNDER 37 C.F.R. § 1.192
U.S. Appln. No.: 09/545,592

16. The apparatus of claim 12, wherein the selected clauses comprise GROUP BY clauses.

17. The apparatus of claim 12, wherein modifying the query comprises:
generating a list of modified queries, wherein each modified query has one or more selected clauses replaced with a false clause; and
executing each modified query until one executes successfully.

18. The apparatus of claim 17, wherein a query executes successfully if it executes without an exception.

19. The apparatus of claim 12, wherein the metadata comprises column type data for the result set.

20. The apparatus of claim 19, further comprising converting the column type data to JAVA types.

21. The apparatus of claim 20, further comprising generating a SQLJ iterator with parameters having the JAVA types.

22. The apparatus of claim 12, further comprising determining whether the query requires a SQLJ iterator.

23. An article of manufacture comprising a computer program carrier readable by computers and embodying one or more instructions executable by the computer for executing a query against a database on a data storage device connected to the computer, comprising:
modifying the query to replace one or more selected clauses with a false clause;

APPELLANTS' BRIEF ON APPEAL
UNDER 37 C.F.R. § 1.192
U.S. Appln. No.: 09/545,592

executing the modified query with the false clause; and
retrieving metadata from the result set obtained by executing the modified query.

24. The article of manufacture of claim 23, wherein the query comprises a SELECT statement.

25. The article of manufacture of claim 24, wherein the SELECT statement is not a SELECT INTO statement.

26. The article of manufacture of claim 23, wherein the selected clauses comprise WHERE clauses.

27. The article of manufacture of claim 23, wherein the selected clauses comprise GROUP BY clauses.

28. The article of manufacture of claim 23, wherein modifying the query comprises:
generating a list of modified queries, wherein each modified query has one or more
selected clauses replaced with a false clause; and
executing each modified query until one executes successfully.

29. The article of manufacture of claim 28, wherein a query executes successfully if it
executes without an exception.

30. The article of manufacture of claim 23, wherein the metadata comprises column
type data for the result set.

31. The article of manufacture of claim 23, further comprising converting the column
type data to JAVA types.

APPELLANTS' BRIEF ON APPEAL
UNDER 37 C.F.R. § 1.192
U.S. Appln. No.: 09/545,592

32. The article of manufacture of claim 31, further comprising generating a SQLJ iterator with parameters having the JAVA types.

33. The article of manufacture of claim 23, further comprising determining whether the query requires a SQLJ iterator.